

# Probabilistic Algorithms for Blind Adaptive Multiuser Detection

Carles Antón-Haro, *Student Member, IEEE*, José A. R. Fonollosa, *Member, IEEE*,  
Zoran Zvonar, and Javier R. Fonollosa, *Senior Member, IEEE*

**Abstract**—In this paper, two probabilistic adaptive algorithms for jointly detecting active users in a DS-CDMA system are reported. The first one, which is based on the theory of hidden Markov models (HMM's) and the Baum-Wech (BW) algorithm, is proposed within the CDMA scenario and compared with the second one, which is a previously developed Viterbi-based algorithm. Both techniques are completely blind in the sense that no knowledge of the signatures, channel state information, or training sequences is required for any user. Once convergence has been achieved, an estimate of the signature of each user convolved with its physical channel response (CR) and estimated data sequences are provided. This CR estimate can be used to switch to any decision-directed (DD) adaptation scheme. Performance of the algorithms is verified via simulations as well as on experimental data obtained in an underwater acoustics (UWA) environment. In both cases, performance is found to be highly satisfactory, showing the near-far resistance of the analyzed algorithms.

## I. INTRODUCTION

RECENTLY, multiuser detection in CDMA systems has received increasing attention [1]. In order to successfully decode the users in a CDMA system, the most important issue to be solved, apart from the multipath propagation problem, is the presence of multiple access interference (MAI). Two approaches can be followed in CDMA scenario: We can apply joint detection (JD) techniques [2], [3] or, otherwise, try to eliminate MAI explicitly (interference cancellation) [4]. In any case, the receiver should know (or should be able to acquire) one or more parameters from the following [5], [6]:

- 1) the signature waveforms of the desired user;
- 2) the signature waveforms of the interfering users;
- 3) the timing (bit-epoch and carrier phase) of the desired user;
- 4) the timing (bit-epochs and carrier phases) of the interfering users;
- 5) the received amplitudes of the interfering users relative to the amplitude of the desired user.

Manuscript received August 28, 1996; revised March 19, 1998. This work was supported in part by the National Plan of Spain, CICYT, under Grants TIC98-0703, TIC98-0412, and TIC96-0500-C10-01, and the Generalitat de Catalunya, CIRIT, under Grant 1996SGR-00096. The associate editor coordinating the review of this paper and approving it for publication was Dr. Petar M. Djuric.

C. Antón-Haro, J. A. R. Fonollosa, and J. R. Fonollosa are with the Department of Signal Theory and Communications, Universitat Politècnica de Catalunya, Barcelona, Spain (e-mail: carles@gps.tsc.upc.es; adrian@gps.tsc.upc.es; fono@gps.tsc.upc.es).

Z. Zvonar is with the Communications Division, Analog Devices, Wilmington MA 01887-1024 USA (e-mail: Zoran.Zvonar@analog.com).

Publisher Item Identifier S 1053-587X(98)07805-2.

The *conventional receiver* neglects the presence of other users requiring only the knowledge of 1) and 3) but, in exchange, it is severely limited by the *near-far* problem. On the other hand, the *optimum detector* [3] attains the best performance by making use of all the aforementioned information and at the expense of exponential computational complexity in the number of users. Among other low-complexity multiuser receivers, we will consider the *decorrelating* [2] and the *MMSE detectors* [7] to show how different levels of knowledge of the parameters can be necessary at the receiver. The former requires knowledge of 1) to 4) to achieve optimum near-far resistance and avoids exponential complexity. The latter is more suited for adaptive implementation on the basis of mean square error (MSE) minimization [1]. In that case, previous knowledge of 2), 4), and 5) can be circumvented by making use of 6) training sequences.

In all cases, external information must be supplied for proper operation of the multiuser algorithms. In the adaptive versions, such information, in the form of training sequences, must be sent not only during the startup period but also after sudden changes in the channel response (CR) or when a new active user appears. An interesting method to detect the latter circumstance is addressed in [8]. The need to retransmit training sequences may be cumbersome in multiuser communications so that, in recent years, a large effort has been made in developing *blind* algorithms that perform CR acquisition and data detection without such information. In other words, only the channel output can be used to obtain estimates on the input data and the CR. In general, blind equalization/estimation methods can be classified in four families [9]:

1. Bussgang-type algorithms [9]–[11];
2. polyspectra and cumulant-based algorithms [9], [12];
3. cyclostationary statistics-based algorithms [13]–[15];
4. probabilistic algorithms [16]–[19].

The algorithms presented and compared in this paper, belong to the fourth group. *Probabilistic algorithms* lead to joint channel estimation and data detection, often on a basis of a maximum likelihood (ML) criterion. These methods exhibit higher computational complexity, but they outperform the other approaches since they make better use of all the known statistical information about the input signal and, in general, require less symbols to obtain an accurate channel estimate.

The proposed algorithms are absolutely blind in the sense that no knowledge of 1) to 6) is required for proper operation. The Viterbi-based algorithm was introduced in [19] to perform jointly blind channel estimation and sequence detection in a

CDMA environment. An extension to multisensor observation was addressed in [20]. On the other hand, the scheme relying on the Baum–Welch (BW) algorithm and the theory of hidden Markov models (HMM's) [21], [22] was proposed in [17] and [18]. The former reference deals with direct application of the batch version of the algorithm, whereas the latter is devoted to develop recursive versions that adaptively estimate the channel parameters and the characteristics of the modulation. However, both studies were developed in a *single-user* framework. The extension of the BW-based algorithm to a multiuser system is addressed in this paper. It is worth noting that the BW-based algorithm leads, at least, to a local maximum of the likelihood function that is not guaranteed by the Viterbi-based one [18], [22].

This paper is organized into eight sections. In Section II, a signal model for a general multiuser digital communications system is presented. Such a model will be utilized in both the BW-based and the Viterbi-based algorithm development. In Section III, the derivation of the BW-based algorithm is carried out. Construction of the HMM, recursive update of the parameter set, inclusion of a blind digital phase-locked loop (DPLL), and strategies to overcome local maxima issues are studied as well. In Section IV, an overview of the Viterbi-based algorithm is given. Section V contains a comparison between both blind algorithms including computational complexity and memory requirements studies. Section VI is devoted to a performance comparison of algorithms in the case of several channel impairments such as the near–far effect, Doppler frequency shifts, low SNR, and asynchronism between users. In addition to simulation study, both algorithms have been tested on experimental data obtained in a UWA channel, and the results are presented in Section VII. Finally, in Section VIII, contributions of this paper as well as directions for future work in this area are commented on.

## II. SIGNAL MODEL

We consider the general asynchronous multiple-access channel model in which the received signal is given by

$$r(t) = \sum_n \sum_{k=1}^K b_k[n] h_k(t - nT) + \sigma w(t) \quad (1)$$

where  $h_k(t - nT)$  is the overall complex channel impulse response of user  $k$  given by the convolution of its signature sequence, physical channel, and receiving filter responses. Its duration is assumed to be no longer than  $L$  symbol periods, and it incorporates the amplitude, delay, and Doppler frequency shifts for user  $k$ . The total number of active users is  $K$ , and their transmitted data sequences are independent binary symbols  $b_k[n] \in \{-1, 1\}$ . The symbol rate is  $1/T$ , and  $w(t)$  is normalized AWGN. The multiple-access channel is sampled at a rate  $f_s = 1/T_s = M/T$  to derive the discrete vector sequence  $\mathbf{r}[n]$

$$\mathbf{r}[n] = [r(nT), \dots, r(nT + (M-1)T_s)]^T \quad (2)$$

where  $^T$  denotes transpose operation. The observation  $\mathbf{r}[n]$  can be modeled as a  $M$ -length vector, probabilistic function

of the state vector  $\mathbf{s}[n]$

$$\mathbf{r}[n] = \mathbf{H}[n]\mathbf{s}[n] + \mathbf{w}[n]. \quad (3)$$

Since, at any given time, a maximum of  $L$  symbols per user affect the observation, there are  $N = 2^{KL}$  possible state vectors corresponding to all combinations of  $L$  binary symbols from the  $K$  active users. We denote each of the possible states as the  $KL$ -length vector  $\mathbf{s}_j$

$$\mathbf{s}_j \in \mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N\} \quad (4)$$

such that

$$\mathbf{s}_j = [\mathbf{s}_{j1}^T, \dots, \mathbf{s}_{jk}^T]^T \quad (5)$$

$$\mathbf{s}_{jk} = [s_{jk}^{(0)}, \dots, s_{jk}^{(L-1)}]^T \quad s_{jk}^{(l)} \in \{-1, 1\}. \quad (6)$$

The actual state at time instant  $nT$  is denoted by  $\mathbf{s}[n]$ ,  $\mathbf{s}[n] \in \mathbf{S}$ .

The  $M \times KL$  matrix  $\mathbf{H}[n]$  depends on the overall discrete channel responses for each user, which is denoted by matrices  $\mathbf{H}_k[n]$

$$\mathbf{H}[n] = [\mathbf{H}_1[n], \dots, \mathbf{H}_K[n]]. \quad (7)$$

Each of these matrices incorporates a vector response for the  $L$  symbols that may be present in the observation due to the intersymbol interference (ISI) or the asynchronous reception

$$\mathbf{H}_k[n] = [\mathbf{h}_{k0}[n], \dots, \mathbf{h}_{k(L-1)}[n]] \quad (8)$$

and, finally, the resulting signature for each user  $k$  and symbol  $l$

$$\mathbf{h}_{kl}[n] = \begin{bmatrix} h_{kl}((n+1)T) \\ \vdots \\ h_{kl}((n+1)T + (M-1)T_s) \end{bmatrix}. \quad (9)$$

The noise is characterized as the  $M$ -length vector  $\mathbf{w}[n]$

$$\mathbf{w}[n] = \sigma[w(nT), \dots, w(nT + (M-1)T_s)]^T. \quad (10)$$

This signal model is common to both the BW- and the Viterbi-based algorithms and will be further refined to meet the formal requirements of each algorithm.

## III. THE BW-BASED BLIND IDENTIFICATION AND DETECTION ALGORITHM

The signal model introduced in the previous section can be used to obtain a description of the received sequence in terms of a HMM [21]. A HMM is a doubly stochastic process with an underlying stochastic process that is not observable (hidden) but can only be observed through another set of stochastic processes that produce the sequence of observed symbols. If a set of statistical parameters (the *model*) is obtainable, it then can be used to identify or recognize other sequences of observations. In the case we are considering, transmitted data constitute the hidden process, whereas the received sequence plays the role of the observable process. These stochastic processes can be characterized by the following parameter set:

- 1) The number of states is  $N = 2^{KL}$ , i.e., the number of distinct inputs that the system may have for a given observation.

- 2) The state transition matrix  $\mathbf{A}$  is given by (11), shown at the bottom of the page, where  $s_{jk}^{(l)} \in \{-1, 1\}$  denotes the  $l$ th symbol (bit) in  $\mathbf{s}_{jk}$ .
- 3) The probability density function of the observation  $\mathbf{r}[n]$  conditioned on a given state channel  $\mathbf{s}_j$  is

$$\mathbf{B} = [b_1(\mathbf{r}[n]), \dots, b_N(\mathbf{r}[n])]^T \quad (12)$$

and

$$b_j(\mathbf{r}[n]) = p(\mathbf{r}[n] | \mathbf{s}_j) = \frac{1}{(\sigma\sqrt{2\pi})^M} \times \exp\left(-\frac{(\mathbf{r}[n] - \mathbf{m}_j[n])^H(\mathbf{r}[n] - \mathbf{m}_j[n])}{2\sigma^2}\right)$$

where

$$\mathbf{m}_j[n] = \mathbf{H}[n]\mathbf{s}_j, \quad 1 \leq j \leq N. \quad (13)$$

The initial state distribution vector  $\pi = [\pi_1, \dots, \pi_N]^T$ , where  $\pi_i = P(\mathbf{s}[0] = \mathbf{s}_i)$ , is assigned an arbitrary value, say,  $\pi_i = 1/N$ . In the sequel, we will refer to such an HMM using the short-hand notation  $\lambda = (\mathbf{A}, \mathbf{B}, \pi)$ .

In this paper, we will assume that  $L$  is either known or can be upper bounded and that the statistics of the transmitter symbols  $P_b = [P(-1), P(1)]^T$ , as well as  $K$ , which is the number of active users contributing to the received signal, are known. The latter assumption is required because the algorithm is blind with respect to the rest of parameters. Actually, if spreading sequences were known, we could check which signatures span the subspace in which the received signal lies (see [8]). This would allow us to estimate the number of active users indirectly. Alternatively, if array observation was taken into account, other array-processing methods could be employed. In addition to the assumptions above and without loss of generality, we will consider that symbols are equally likely since this allows us to simplify the derivations and the resulting expressions for the proposed estimation methods. Regarding the *unknown* parameters in our model, we can follow standard HMM-based approaches to estimate them. To be precise, the parameters to be estimated are

- 1)  $\mathbf{M}$ , which is the  $M \times N$  means matrix given by

$$\mathbf{M}[n] = [\mathbf{m}_1[n], \dots, \mathbf{m}_N[n]] \quad (14)$$

which stands for the noise-free ISI-corrupted received multiuser signal corresponding to each state of the model; note that according to Fig. 1, all  $K$  users contribute to the means vector in the following manner:

$$\mathbf{m}_j[n] = \sum_{k=1}^K \mathbf{m}_{jk}[n]. \quad (15)$$

- 2) the AWGN variance  $\sigma^2$ .

#### A. The Baum–Welch (BW) Identification Algorithm

This iterative batch procedure is essentially identical to the expectation-maximization (EM) method, and it is known to lead, at least, to a local maximum of the likelihood function (e.g. [22]).

Let us consider a block of  $M \cdot D$  samples of the received sequence  $\mathbf{r}_D = [\mathbf{r}[1]^T, \mathbf{r}[2]^T, \dots, \mathbf{r}[D]^T]^T$ . First of all, we will obtain  $\gamma_i[n]$  ( $n = 1 \dots D$ ), i.e., the probability of being in state  $\mathbf{s}_i$  at time  $n$ , given  $\mathbf{r}_D$  and the model by means of the computationally efficient *forward–backward* algorithm. Second, we will re-estimate the parameters of the model using the BW reestimation formulas. This scheme is iterated until convergence to a stable point is attained. For our model, the resulting BW iteration is as follows:

- S1) *Compute the normalized forward variable*  $\alpha_i[n] = P(\mathbf{r}[1], \mathbf{r}[2], \dots, \mathbf{r}[n], \mathbf{s}[n] = \mathbf{s}_i | \lambda)$ , i.e., the probability of the partial observation sequence until time instant  $n$  and state  $\mathbf{s}_i$ , at time  $n$ , given the model  $\lambda$ .

• *Initialization:*

$$\alpha_i[1] = \frac{1}{N} p_i(\mathbf{r}[1]), \quad 1 \leq i \leq N \quad (16)$$

- *Induction steps:*  $\alpha_j[n+1]$  is computed on the basis of  $\alpha_i[n]$  and the transition probability matrix. The scale factor  $c[n]$  is used in order to prevent the algorithm from underflow by normalizing the forward and backward variables.

for  $n = 1, 2, \dots, D-1$  do

$$c[n] = \left( \sum_{i=1}^N \alpha_i[n] \right)^{-1} \quad (17)$$

$$\alpha_j[n+1] = c[n] b_j(\mathbf{r}[n+1]) \sum_{i=1}^N \alpha_i[n] a_{ij}, \quad 1 \leq j \leq N \quad (18)$$

end.

- *Termination:*

$$c[D] = \left( \sum_{i=1}^N \alpha_i[D] \right)^{-1} \quad (19)$$

- S2) *Compute the normalized backward variable*  $\beta_i[n] = P(\mathbf{r}[n+1], \mathbf{r}[n+2], \dots, \mathbf{r}[D] | \mathbf{s}[n] = \mathbf{s}_i, \lambda)$ , i.e., the probability of the partial observation sequence from  $n+1$  to the end, given the state  $\mathbf{s}_i$  at time  $n$  and the model  $\lambda$ :

• *Initialization:*

$$\beta_i[D] = c[D], \quad 1 \leq i \leq N \quad (20)$$

$$\mathbf{A} = \{a_{ij}\}, \quad 1 \leq i, j \leq N \quad (11)$$

$$a_{ij} = P(\mathbf{s}[n+1] = \mathbf{s}_j | \mathbf{s}[n] = \mathbf{s}_i) = \begin{cases} P(s_{j1}^{(0)} \cap \dots \cap s_{jK}^{(0)}) = 2^{-K}, & \text{if } s_{jk}^{(l)} = s_{ik}^{(l-1)}, \quad l = 1, \dots, L-1, \quad k = 1, \dots, K \\ 0, & \text{otherwise} \end{cases}$$

- *Induction steps:*

for  $n = D - 1, \dots, 1$  do

$$\beta_i[n] = c[n] \sum_{j=1}^N a_{ij} b_j(\mathbf{r}[n+1]) \beta_j[n+1], \quad 1 \leq i \leq N \quad (21)$$

end.

S3) Compute  $\gamma_i[n]$ , which is the probability of being in state  $\mathbf{s}_i$  at time  $n$ , given  $\mathbf{r}_D$  and the model.

$$\gamma_i[n] = \alpha_i[n] \beta_i[n], \quad 1 \leq i \leq N, \quad 1 \leq n \leq D. \quad (22)$$

S4) Re-estimate the parameters of the model using the BW reestimation formulas. For each state, the mean is computed 1) weighting every observation with the probability of being in such state and 2) averaging over the whole received sequence

$$\hat{\mathbf{m}}_i = \frac{\sum_{n=1}^D \gamma_i[n] \mathbf{r}[n]}{\sum_{n=1}^D \gamma_i[n]}, \quad 1 \leq i \leq N. \quad (23)$$

The estimate for the variance of the noise is derived in a similar manner as

$$\hat{\sigma}^2 = \frac{1}{MD} \sum_{n=1}^D \sum_{i=1}^N \gamma_i[n] \|\mathbf{r}[n] - \hat{\mathbf{m}}_i\|^2. \quad (24)$$

S5) Repeat the above four steps until convergence to a stable point is achieved.

Afterwards, data detection can be performed following, for example, a *maximum a posteriori* (MAP) criterion. At instant  $n$ , the index to the MAP state is

$$i_n = \arg \max_{1 \leq i \leq N} \{\gamma_i[n]\} \quad (25)$$

and consequently, an estimate for the transmitted data symbols  $\hat{\mathbf{b}}[n] = [\hat{b}_1[n], \dots, \hat{b}_K[n]]^T$  can be obtained by properly selecting among the elements of the MAP state  $\mathbf{s}_{i_n}$ . However, in the event that there are disallowed transitions (i.e.,  $a_{ij} = 0$  for some  $i, j$ ), the resulting state sequence cannot be guaranteed to be a valid sequence. Applying a most likely state sequence criterion overcomes this problem at the expense of higher computational complexity. Since, in practice, the problem outlined for the MAP criterion does not usually occur [22], we chose, for the sake of simplicity, the first criterion to carry out sequence detection. Note, however, that data detection is a side process in the estimation loop since detected bits are not used in the re-estimation formulas except for  $\gamma_i[n]$ . Hence, this simplification does not degrade convergence properties of the algorithm.

### B. Recursive Estimation of the CR

The procedure described in the previous subsection is *batch type* in the sense that it is iterated for the *whole* received sequence until convergence (at least to a local maxima) is achieved. Moreover, it implicitly assumes that the unknown parameters remain unchanged along the received sequence ( $\mathbf{m}_i[n] = \mathbf{m}_i, n = 1 \dots D$ ). When dealing with time-varying

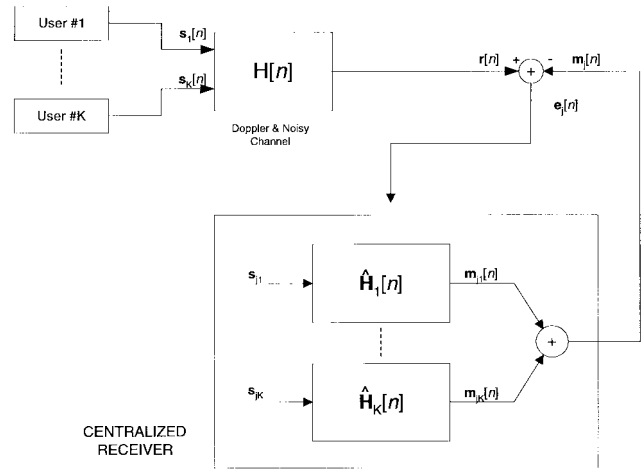


Fig. 1. System model.

channels, some modifications must be introduced. For instance, in [24], the time-varying nature of the CR is included in the batch BW re-estimation formulas. Another approach is considered in [18], where an adaptive version of the above-mentioned algorithm called the adaptive Baum-Welch (ABW) is presented. In this case, the BW re-estimation formulas in (23) and (24) are rewritten in a recursive manner so that the algorithm can track slow variations of the parameters of the model.

Whenever the channels are modeled as linear FIR systems, another strategy can be adopted. Instead of adapting the means, we can directly adapt the CR estimate. In the framework of the steepest-descent algorithm for system identification, we have (see Fig. 1)

$$\hat{\mathbf{H}}[n] = \hat{\mathbf{H}}[n-1] + \mu_h E[\mathbf{e}[n] \mathbf{s}[n]^H] \quad (26)$$

where

$$\mathbf{e}[n] = \mathbf{r}[n] - \mathbf{m}[n] = \mathbf{r}[n] - \mathbf{H}[n] \mathbf{s}[n] \quad (27)$$

$$\mathbf{s}[n] = [\mathbf{s}_1[n]^T, \dots, \mathbf{s}_K[n]^T]^T \quad (28)$$

and  $\mu_h$  is the adaptation constant. The CR matrix in the new parameter set  $(\hat{\mathbf{H}}, \hat{\sigma}^2)$  will be related to the means matrix through  $\hat{\mathbf{M}}[n] = \hat{\mathbf{H}}[n] \hat{\mathbf{S}}$ . In our blind environment, the expectation in the gradient term will be computed, again on the basis of the sequence and the model at instant  $n-1$

$$\begin{aligned} \hat{\mathbf{H}}[n] &= \hat{\mathbf{H}}[n-1] + \mu_h E_{r_{n+\Delta}, \lambda_n}[\mathbf{e}[n] \mathbf{s}[n]^H] \\ &= \hat{\mathbf{H}}[n-1] + \mu_h \sum_{j=1}^N \gamma_j[n] \mathbf{e}_j[n] \mathbf{s}_j^H \end{aligned} \quad (29)$$

where  $\mathbf{e}_j[n]$  is defined as

$$\mathbf{e}_j[n] = \mathbf{r}[n] - \hat{\mathbf{m}}_j[n], \quad 1 \leq j \leq N. \quad (30)$$

The AWGN variance estimate is also recursively computed in accordance with

$$\hat{\sigma}^2[n] = \hat{\sigma}^2[n-1] + \mu_s \left( \frac{1}{M} \sum_{j=1}^N \gamma_j[n] \|\mathbf{e}_j[n]\|^2 - \hat{\sigma}^2[n-1] \right). \quad (31)$$

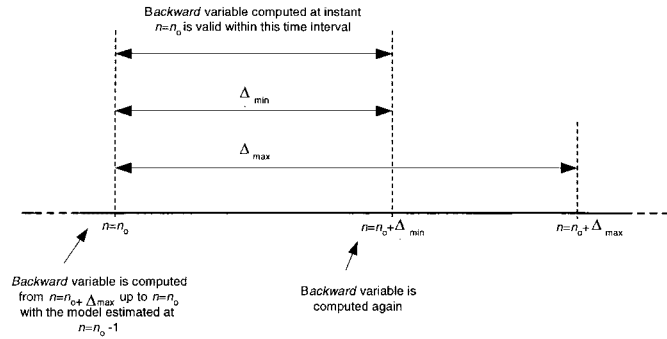


Fig. 2. Computing backward variable in a sawtooth style.

Note that  $\hat{\sigma}^2$  will *asymptotically* attain the estimate of the AWGN variance, but in the first iterations (symbols) when the estimate of  $\mathbf{H}$  is still coarse, this estimation error will also revert in  $\hat{\sigma}^2$ . The importance of this fact will be discussed in Section III-D.

The initialization and time-recursive computation of the normalized forward variable are still valid for the adaptive version of the algorithm. For the on-line computation of the (normalized) backward variable, we can follow a fixed-lag scheme or the computationally more efficient sawtooth-lag scheme (Fig. 2) [25]. In the fixed-lag case, we choose a lag  $\Delta$  and *apply for each  $n$  the backward recursion S2 from  $n + \Delta$  to  $n$* . In the sawtooth-lag scheme, which is the one considered in this paper, we work with a lag varying from a fixed lag  $\Delta_{\min}$  to another  $\Delta_{\max}$ . Hence, we do not need to perform the backward recursion for each symbol, but we do for each interval of  $\Delta_{\min}$  symbols. The equivalent expression for  $\Delta$  in that case is

$$\Delta_{\text{eq}} = \frac{\Delta_{\max}}{\Delta_{\min}}. \quad (32)$$

Rapid changes in the CR are usually addressed by employing an RLS updating scheme that exhibits better tracking properties than the proposed LMS-based approach. In principle, the RLS update equations could be rewritten in a *blind* manner, i.e., including variable  $\gamma_i[n]$ , to compensate for the lack of knowledge about training sequences. This approach, however, has not been considered in this work due to the associated high computational complexity.

### C. Blind DPLL

In order to increase the robustness of the receiver in environments with large frequency deviations,<sup>1</sup> we include specific mechanisms to estimate and track the resulting phase shifts. As depicted in Fig. 3, the CR estimate for each user can be factored in two terms, namely

$$\hat{\mathbf{H}}_k[n] = e^{j\hat{\theta}_k[n]} \cdot \hat{\mathbf{H}}_k[n]. \quad (33)$$

The phase-corrected term of the overall CR ( $\hat{\mathbf{H}}_k[n]$ ) will be adaptively estimated using the steepest-descent procedure described in the previous section, whereas the phase term  $\theta_k[n]$  will be updated using a *blind* DPLL [26]. It is obtained by

<sup>1</sup> Such frequency deviations appear as a consequence of vehicles' speed or other propagation-related effects.

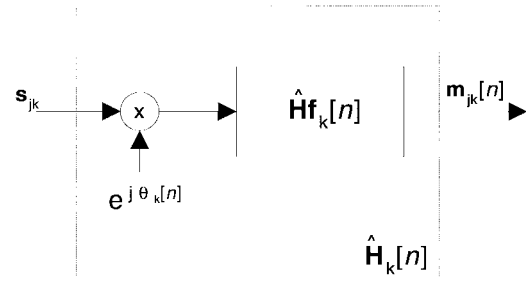


Fig. 3. Inclusion of a blind DPLL.

recognizing that the gradient of the MSE with respect to the phase estimate represents the output of an equivalent phase detector (c.f., [27]).

To obtain the necessary tracking capabilities, and according to [23], a second-order scheme should be used. The update equation, particularized for user  $k$ , is given by

$$\hat{\theta}_k[n] = \hat{\theta}_k[n-1] + k_1 \phi_k[n] + k_2 \sum_{m=0}^{n-1} \phi_k[m] \quad (34)$$

where  $k_1$  and  $k_2$  are proportional and integral tracking constants, respectively, and  $\phi_k[n]$  is the phase detector output. Based on the analogy with the digital PLL, we replace  $\phi_k[n]$  with the MSE gradient w.r.t. the phase estimate for user  $k$ , i.e.,

$$\phi_k[n] = \nabla_{\theta_k}(E[\|\mathbf{e}[n]\|^2]) = \Im\{E[\mathbf{e}[n]^H \mathbf{m}_k[n]]\} \quad (35)$$

where  $^H$  stands for conjugate transpose, and transmitted data sequences have been assumed to be uncorrelated. In our blind environment, the expectation in the gradient term is computed on the basis of the received data and the model at instant  $n-1$ , namely

$$\begin{aligned} \phi_k[n] &= \Im\{E_{\mathbf{r}_{n+\Delta}, \lambda_n}[\mathbf{e}[n]^H \mathbf{m}_k[n]]\} \\ &= \Im\left\{\sum_{j=1}^N \gamma_j[n] \mathbf{e}_j[n]^H \hat{\mathbf{m}}_{jk}[n]\right\} \end{aligned} \quad (36)$$

where  $\mathbf{e}_j[n]$  is defined as in (30).

Arranging the elements of (34) in a column vector, we get the vector expression

$$\hat{\boldsymbol{\theta}}[n] = \hat{\boldsymbol{\theta}}[n-1] + K_1 \boldsymbol{\phi}[n] + K_2 \sum_{m=0}^{n-1} \boldsymbol{\phi}[m] \quad (37)$$

which reflects DPLL operation. It is worth noting that since the DPLL is *blind*, phase tracking can be performed up to a phase constant, being that this limitation is inherent to any blind detection scheme.

Finally, note that such phase estimates should be included in the update equations for  $\hat{\mathbf{H}}_k[n]$ . Thus, (29) turns into

$$\hat{\mathbf{H}}_k[n] = \hat{\mathbf{H}}_k[n-1] + \mu_h \sum_{j=1}^N \gamma_j[n] \mathbf{e}_j[n] (\mathbf{s}_j \odot e^{j\hat{\theta}_k[n] \otimes \mathbf{1}_L})^H \quad (38)$$

where  $\odot$  and  $\otimes$  stand, respectively, for the Hadamard and Kronecker products, and  $\mathbf{1}_L$  is an all-ones  $L \times 1$  vector.

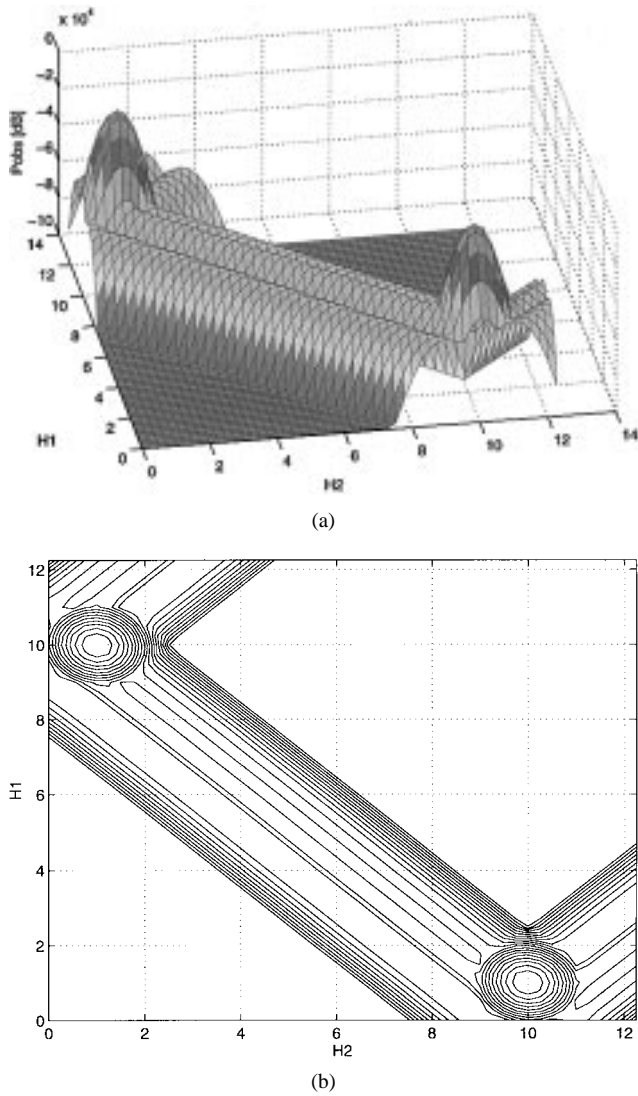


Fig. 4. (a) Likelihood function and (b) contour plot (bottom) for the MABW algorithm. Users' amplitudes: 10, 1.

#### D. Avoiding Local Maxima

As mentioned previously, BW-based algorithms lead at least to a local maximum of the likelihood function. When dealing with CDMA signals, those maxima appear as a consequence of two fairly independent phenomena: 1) severe ISI due to multipath propagation and asynchronism between users and 2) the fact that the multiuser signal is affected by the *near-far* effect. The first circumstance rarely occurs in channels with low to moderate ISI. Therefore, we will focus on the mechanisms to overcome the second phenomenon.

When the received signal is affected by the near-far effect, the algorithm tends to split the CR estimate of the user(s) of greatest magnitude among the weakest(s) ones since it locally maximizes the likelihood function. Nevertheless, this extent can be easily detected since, in those cases, estimated data sequences for the involved users are identical (up to a sign change). In Fig. 4, the existence of local maxima is illustrated for a (simplified) case in which two users with amplitudes 10 and 1 and the same spreading sequence contribute to the received signal. Hence, users are no longer orthogonal but,

as long as the MABW algorithm is essentially a *clustering* algorithm [see (23)], the only requirement for proper operation is that separation among clusters is large enough w.r.t. the additive noise level. Obviously, this condition is satisfied if users' amplitudes are dissimilar.

In the above-mentioned figures, the logarithm of the probability of observation of the received sequence  $\log[p(\mathbf{r} | \lambda)] = \log[\Pr(\mathbf{r}[1], \dots, \mathbf{r}[N] | \lambda)]$  (i.e., the log-likelihood function), as well as the corresponding contour plot, are shown.<sup>2</sup> The log-likelihood function exhibits two global maxima: the true solution, which is located at (10, 1), and symmetric solution at (1, 10). That is, the algorithm leads to an arbitrary assignment of the estimated CR's to the active users, as in any blind approach. Furthermore, the straight lines between the global maxima constitute the locus of the local maxima arising from the arbitrary splitting of the estimate for the strongest CR among the rest of users.

The strategy considered in this paper to overcome this problem is referred to as *coherence checking* and was suggested in [20]. It consists of checking, every  $N_c$  symbols, the crosscorrelation of the estimated data sequences among the  $K$  active users (each of them being evaluated for  $2 \cdot L + 1$  lags). The probability of random coincidence of two independent binary sources, except for a possible sign change, is  $1/2^{N_c-1}$  and can be made arbitrarily small increasing  $N_c$ . In case this absolute coherence is detected, it is assumed that the algorithm has converged to a local minimum and the impulse responses of the, say,  $K_m$  matching users are added (or subtracted if the sequences were of opposite sign) and assigned arbitrarily to one of them. The estimated impulse responses of the remaining  $K_{m-1}$  users are reset to the initial value (at  $n = 0$ ). More formally, the cross-correlation functions are defined as

$$\rho_{kk'}|l| = \left[ \frac{1}{N_c - |l|} \sum_{n=0}^{N_c-1} \hat{b}_k[n+l] \hat{b}_{k'}[n] \right]; \quad k' > k \quad (39)$$

where  $\hat{b}_k[n]$  is the estimated ML sequence associated with user  $k$  for the previous  $N_c$  symbols. The global impulse response is compensated by iterating, for each user in ascending order, i.e.,  $k = 1, \dots, K$ , the following impulse response update:

$$\hat{\mathbf{h}}_{kl}[n] \leftarrow \hat{\mathbf{h}}_{kl}[n] + \sum_{k'=k+1}^K \sum_{l'=0}^{L-1} \rho_{kk'}[l-l'] \hat{\mathbf{h}}_{k'l'}[n] \quad (40)$$

and possible reinitialization

$$\begin{aligned} \hat{\mathbf{h}}_{k'l'}[n] &\leftarrow |\rho_{kk'}[l-l']| \hat{\mathbf{h}}_{k'l'}[n=0] \\ &\quad + (1 - |\rho_{kk'}[l-l']|) \hat{\mathbf{h}}_{k'l'}[n] \\ k' &= k+1, \dots, K \quad l' = 0, \dots, L-1. \end{aligned} \quad (41)$$

Consequently, in high near-far scenarios, users tend to be sorted with descending power. Those compensations may be necessary until a good estimate of the *strong* impulse response is obtained, which cancels the local maxima.

Regarding other identifiability issues, time-shift ambiguity is observed as in any blind procedure where no temporal

<sup>2</sup> According to the definition of the forward variable, the likelihood function can be efficiently computed by adding the value of the forward variable at time instant  $N$ . See [22] for details.

reference is given to the receiver. On the contrary, scale and phase ambiguity is not observed in our algorithm except for a  $180^\circ$  phase shift inherent to any blind scheme applied to a  $180^\circ$  rotation-invariant constellation.

As a final remark, note that the AWGN variance estimate is fed back into the algorithm, supplying information on how *noisy* the environment is. Consequently, special attention must be paid to the fact that at any moment, the level of the already-estimated CR's has to be kept above this magnitude. Otherwise, the algorithm would consider such user(s) as a part of the noise and would give up estimating their associated CR's. An increase in the AWGN variance estimate would follow, which, in turn, would affect stronger users; eventually, the system would be driven toward the trivial solution ( $\hat{\mathbf{H}} \equiv 0$ ), i.e., considering the whole signal as noise. This might happen, for instance, after reassignments of the estimated CR's resulting from coherence checking.

#### E. Switching to a Decision-Directed Mode

As mentioned at the beginning of this paper, it is intended to switch to a DD mode of operation once convergence has been attained in the blind stage. We show in Section VI (simulation analysis) that the CR estimate obtained with the blind algorithm is *good* enough to make such a change feasible. Although adaptive versions of ML algorithms can be considered for this DD stage [28], we chose, for the sake of simplicity, a linear detector updated with a MMSE strategy [5]. The gap between estimation and equalization can be covered by obtaining the initial values for equalizer taps from the last CR estimate in the blind stage [29].

#### F. Summary of the Algorithm

To summarize, the blind algorithm results in the following:

- *Initialization:*  
Set initial values:  $\hat{\mathbf{H}}\mathbf{f}[0]$ ,  $\hat{\theta}[0]$  and  $\hat{\sigma}^2[0]$ .
- *Iteration for all received data:*  
for  $n = 1 \dots D$ ,
  - ◊ *Computation of  $\hat{\mathbf{M}}[n]$*
  - ◊ *ABW iteration:*
    - \* Forward-Backward iteration, which allows us to compute  $\gamma_i[n]$  ( $i = 1 \dots N$ ) from  $\hat{\mathbf{M}}[n]$  and  $\hat{\sigma}^2[n]$ .
    - \* Recursive reestimation of the parameter set:  $\hat{\mathbf{H}}\mathbf{f}[n]$ ,  $\hat{\theta}[n]$  and  $\hat{\sigma}^2[n]$ .
    - \* Data detection (MAP criterion)
  - ◊ *Coherence check* (every  $N_c$  symbols)

end;

From now on, this algorithm will be referred to as the multiuser adaptive Baum-Welch (MABW) algorithm.

### IV. OVERVIEW OF THE VITERBI-BASED BLIND IDENTIFICATION AND DETECTION ALGORITHM

This blind estimation algorithm is thoroughly explained in [19] and [20]. Therefore, only its major features will be outlined in the following paragraphs.

The algorithm is initialized with an arbitrary CR estimate for each state in the model. Note that now, such CR matrix

contains as many matrices defined in (7) as the number of states, namely, the  $M \times KLN$  matrix

$$\begin{aligned} \mathcal{H}_j[n] &= \mathbf{H}[n]_{\text{state } j} \quad j = 1, \dots, N \\ \mathcal{H}[n] &= [\mathcal{H}_1[n], \dots, \mathcal{H}_N[n]]. \end{aligned} \quad (42)$$

Similarly, vector  $\boldsymbol{\theta}[n]$  turns into a  $K \times N$  matrix  $\boldsymbol{\Theta}[n]$  as well:

$$\begin{aligned} \boldsymbol{\Theta}_j[n] &= \boldsymbol{\theta}[n]_{\text{state } j} \quad j = 1, \dots, N \\ \boldsymbol{\Theta}[n] &= [\boldsymbol{\Theta}_1[n], \dots, \boldsymbol{\Theta}_N[n]]. \end{aligned} \quad (43)$$

The algorithm is then iterated at the symbol rate as in a conventional Viterbi scheme. Metrics along the trellis are computed using, for each state, its own current *estimates* of the overall CR. At each iteration, a ML path is continued for each state selecting the most likely possible predecessor and updating the accumulated metric. Each state shares  $L - 1$  symbols, for each user, with its set of possible predecessors, and thus, there are  $2^K$  predecessors per state.

The phase-corrected channel estimate matrix for each state  $\mathcal{H}f_j$  is updated by modifying the one associated with its most likely predecessor using the standard stochastic gradient LMS scheme

$$\hat{\mathcal{H}}f_j[n] = \hat{\mathcal{H}}f_i[n-1] + \mu_h \mathbf{e}_{ij}[n] (\mathbf{s}_j \odot e^{j\hat{\theta}_j[n] \odot \mathbf{1}_L})^H \quad (44)$$

where the state  $\mathbf{s}_i$  is the predecessor to state  $\mathbf{s}_j$ , and the error vector  $\mathbf{e}_{ij}[n]$  is defined as

$$\mathbf{e}_{ij}[n] = \mathbf{r}[n] - \hat{\mathcal{H}}f_i[n-1] (\mathbf{s}_j \odot e^{j\hat{\theta}_j[n] \odot \mathbf{1}_L}). \quad (45)$$

The phase correction to compensate for Doppler frequency shift is updated, as before, using a second-order DPLL [see (34)–(37)]. Again, the expectation in the phase gradient term in (35), for each present state, is replaced by a stochastic estimate involving only the ML predecessor and the present state itself.

After convergence, the channel response corresponding to the most likely final state can be employed to yield the ML transmitted sequence for each user, as in the conventional multiuser optimal receiver. In the sequel, this algorithm will be referred to as the multiuser adaptive Viterbi (MAV) algorithm.

### V. COMPARISON BETWEEN ALGORITHMS

In this section, we will point out differences and similarities between both algorithms in terms of how blind identification is performed. We will also study and compare their computational complexity. Finally, memory requirements will be analyzed.

#### A. Algorithmic Comparison

Algorithmic differences between both algorithms can be classified in three groups: regarding the structure of the parameter set, the manner in which this parameter set is updated, and the method to perform data detection.

- Structure of the Parameter Set: First of all, note that the MAV algorithm keeps track of an estimate of the CR (phase-corrected CR plus phase correction term) for *each state* of the model, whereas the MABW algorithm just holds one averaged CR estimate.

As for the estimate of  $\sigma^2$ , it is worth noting that there is no need to obtain such an estimate when considering the MAV algorithm. The reason is that such an algorithm operates directly with the metrics instead of with

TABLE I

COMPUTATIONAL COMPLEXITY OF DIFFERENT TASKS (ORDER OF MAGNITUDE) EXPRESSED IN TERMS OF OPERATIONS PER SYMBOL INTERVAL ( $K$ : NR. OF USERS,  $L$ : DELAY SPREAD IN SYMBOLS,  $M$ : CHIPS/SYMBOL,  $N = 2^{KL}$ : NR. OF STATES)

Section	BW-based	Viterbi-based
Doppler correction	$N \cdot K \cdot L$	
Computation of the means	$N \cdot K \cdot L \cdot M$	
Computation of $\gamma$	$(\Delta_{eq}+1) \cdot N^2 + N \cdot M$	—
Computation of the metrics	—	$N^2 + N \cdot M$
Reestimation of $\sigma^2$	$N \cdot M$	---
CR reestimation	$N \cdot K \cdot L \cdot M$	
Doppler reestimation	$N \cdot K \cdot M$	
Coherence check	$K^2 \cdot L$	

the whole pdf as specified in (12). This is one of the advantages of the Viterbi-based approach in which no exponent operation must be performed.

- **Update of the Parameter Set:** Both algorithms employ the same schemes for updating the parameters of the model (gradient algorithms for the CR and  $\sigma^2$  estimates, and a second-order DPLL for the phase-correction estimate). Nevertheless, in the case of MABW, the new estimate for  $\mathbf{Hf}$ ,  $\theta$ , and  $\sigma^2$  (which are shared by all  $N$  states) is obtained by averaging the error for every state weighted with  $\gamma_i[n]$  [see (29), (31), and (36)], i.e., following the steepest-descent approach. On the contrary, only the error with respect to the most likely predecessor state is considered in the MAV algorithm (stochastic gradient LMS).
- **Data Detection:** Again, data detection is performed differently by both algorithms. The MLSE criterion is followed by the Viterbi-based algorithm, whereas the MAP criterion is adopted in the other case. Consequently, the MABW algorithm cannot guarantee the sequence of states being a valid one. Nevertheless, it is not a key issue since these blind identification algorithms are thought mainly as an alternative to the use of training sequences to acquire a reliable initial estimate of the parameter set. Data detection is to be performed basically by using classical approaches once convergence has been attained.

### B. Computational Complexity

In Table I, the computational complexity for different sections of both algorithms is reported and compared. Two considerations should be made. First, such complexity is expressed in terms of magnitude order. This is due to the dropping of multiplicative constants as well as additive terms that are much less significant than those reported. Moreover, contributions from sums, products, exponents, etc., with complex and/or real data are assumed to be the same. Second, computational complexity related with data detection is not considered in any event since it is negligible in comparison with that of other sections.

It can be observed that the section contributing the most is that of the computation of  $\gamma_i[n]$  in the case of the MABW algorithm, whose counterpart for MAV is the computation of the metrics. Both contribute the same to the total number of operations [ $O(N^2)$ ] up to a scale factor of  $(\Delta_{eq} + 1)$  in the case of the MABW algorithm resulting from the need of recursively (re)computing the backward variable. However,

it is not an important drawback since this factor can be significantly reduced with the use of the sawtooth-lag scheme, where  $\Delta_{eq}$  can be forced to be approximately equal to 1. It should also be noted that  $N^2$  is, in fact, an upper bound. As long as the transition probabilities  $a_{ij}$  are known and are different from zeros in  $2^K$  cases among  $N = 2^{KL}$ , an efficient implementation for those algorithms would only require on the order of  $N \cdot 2^K$  operations per symbol. Regarding the rest of sections, they all contribute on the order of  $O(N)$  or less and equally for both algorithms.

Let us consider the following numerical example: a system with  $K = 4$  users whose signatures are Gold sequences of  $M = 7$  chips/symbol and ISI spanning along  $L = 2$  symbols. It follows that the number of states in the model is  $N = 2^{KL} = 256$ . Under these conditions and assuming that the backward variable is computed according to a sawtooth-lag scheme with  $\Delta_{eq} = 1.5$ , both algorithms would require on the order of  $10^5$  operations/symbol.

For comparison purposes, we will refer the algorithms' complexity to that of an *adaptive* MLSE scheme in which not only the metrics must be obtained, but a *single* CR is adaptively estimated as well. This approach involves introducing a delay such that the detected symbols obtained from the backtracking in order to update the CR estimate are reliable [28]. Hence, the numerical burden associated with the computation of the metrics (or, equivalently, computation of  $\gamma_i[n]$ ) is approximately the same, whereas computations associated with CR estimation make the difference. As a matter of fact, in the blind approaches,  $N$  channel estimates must be obtained at each time instant in order to keep  $N$  separate estimates (MAV) or, conversely, a single averaged channel estimate (MABW). In any event, computational complexity is increased by a factor of  $N$  with respect to the classical Viterbi algorithm.

One strategy to reduce the computational burden would consist of defining, from the  $N = 2^{KL}$ -state HMM, an *aggregated* Markov chain with a reduced number of  $N' = 2^K$  states [30]. In general, this can be accomplished by considering the symmetries in the transition matrix. In the model we are considering, those symmetries arise from the fact that the  $N$ -state Markov chain is an FIR-filtered version of the  $N'$ -state chain.

As a final conclusion, the computational complexity of both strategies can be said to be approximately the same for a reduced number of users and low delay spread and on the same order of magnitude in any other event. Nevertheless, the exponential dependence of computational complexity on both  $K$  and  $L$  would prevent the algorithms from working in real time even for a moderate number of active users in the case of significant ISI.

### C. Memory Requirements

Whereas the MABW algorithm is (slightly) more costly in terms of number of operations per symbol, MAV demands more memory resources. This is a direct consequence of keeping track of CR's separately (i.e., one estimate for each state), which increases memory requirement exponentially in  $K$  and  $L$ . Table II reflects this fact. The term *storage units* is employed to describe uniformly the amount of memory necessary to store a unit of any type of data (real or complex).



TABLE II

MEMORY REQUIREMENTS (ORDER OF MAGNITUDE) EXPRESSED IN TERMS OF STORAGE UNITS ( $K$ : NR. OF USERS,  $L$ : DELAY SPREAD IN SYMBOLS,  $M$ : CHIPS/SYMBOL,  $N = 2^{KL}$ : NR. OF STATES)

Estimated parameter	BW-based	Viterbi-based
Doppler ( $\Theta$ )	$K$	$N \cdot K$
CR ( $H_f / \mathcal{H}_0$ )	$K \cdot L \cdot M$	$N \cdot K \cdot L \cdot M$
AWGN variance ( $\sigma^2$ )	1	—
Means ( $\mathbf{M}$ )	—	$N \cdot K \cdot M$
$\gamma$	$N$	—
Metrics	—	$N$
State matrix ( $\mathbf{S}$ )	—	$N \cdot K \cdot L$
Transition matrix ( $\mathbf{A}$ )	—	$N^2$

Again, since transition matrix  $\mathbf{A}$  is *sparse* (especially for large  $L$ ), significant reduction in the requirements can be achieved only in those positions where  $a_{ij} = 2^{-K} \neq 0$  are stored.

In the conditions of the example in the last section, on the order of  $2 \cdot 10^4$  storage units would be required in the first case, in contrast to  $5 \cdot 10^4$  in the second when storing  $\mathbf{A}$  in a compressed manner.

## VI. COMPARATIVE PERFORMANCE STUDY: SIMULATION ANALYSIS

The system under study operates at the symbol rate employing a BPSK modulation scheme. In all cases,  $K = 4$  users contributed to the CDMA signal. Gold sequences with  $M = 7$  chips were used as spreading sequences; coherence checks were performed every  $N_c = 50$  symbols. A stationary single-path model was considered for the channel. For all simulations, the incoming signal was sampled at the chip rate, and the constants in the update equations were set to  $\mu_h = 2 \cdot 10^{-2}$ ,  $\mu_s = 4 \cdot 10^{-2}$ ,  $k_1 = 10^{-3}$ , and  $k_2 = 10^{-4}$ . Several tests were performed to determine the power difference between users up to 30 dB and for SNR of the weakest user, ranging from 6 to 12 dB. Distortion due to Doppler frequency shift was considered as well. Unless otherwise stated, simulation results are averaged over 50-run tests.

### A. General Behavior

General behavior for both algorithms is illustrated in Figs. 3–5, with a case consisting of  $K = 4$  users received with similar power (amplitudes: 3, 2, 1, 1). Fig. 5(a) shows the evolution of the CR estimate (real part) obtained with the MABW algorithm, whereas Fig. 5(b) is devoted to plotting the evolution of such a parameter obtained with the MAV (one-run plots). As a complement to those figures, the averaged estimation noise—learning curves—for all users and both algorithms are plotted in Fig. 6. Such parameters are computed according to

$$\xi_k[n] = \frac{1}{ML} \sum_{m=0}^{M-1} \sum_{l=0}^{L-1} \left\| \hat{h}_{kl}((n+1)T + mT_s) - h_{kl}((n+1)T + mT_s) \right\|^2$$

$$k = 1 \dots K \quad n = 1 \dots D. \quad (46)$$

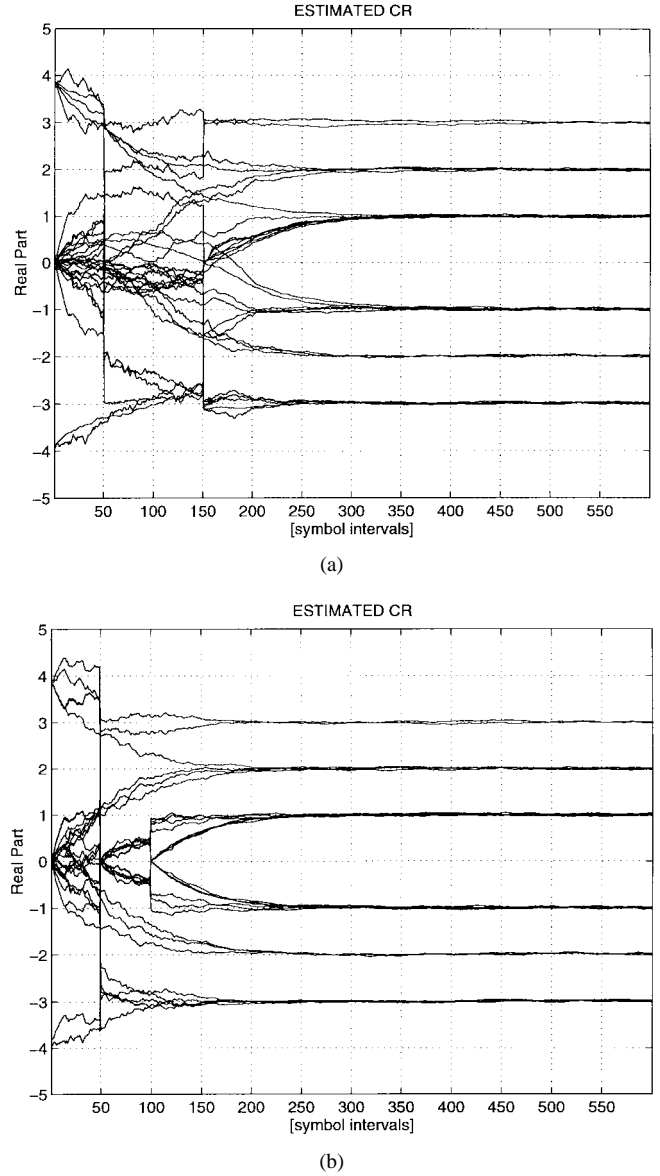


Fig. 5. Evolution of the CR estimates (real part) in one-run tests: (a) MABW and (b) MAV. Users' amplitudes: 3, 2, 1, 1. SNR = 12 dB.

First, a slower convergence for the BW-based algorithm is clearly observed. This fact is a direct consequence of the way the CR estimates are obtained: An *average* over all the possible present states is used by the MABW algorithm, whereas, in the other case, only the preceding and present states are involved in the update of the CR estimate corresponding to each state. Within the first iteration steps, when there is not a predominant path in the trellis, all states contribute to updating the CR estimate; this extent guarantees convergence at least toward a local maximum but, of course, brings down convergence speed. Additionally, it should be pointed out that the AWGN variance estimate obtained with such an algorithm affects the estimate of the CR through (12), disallowing either an excessively fast evolution or coherence detection until the estimation noise decreases to safe levels. For instance, note that the range in which coherence is detected and CR's are rearranged (sudden jumps in the estimation noise

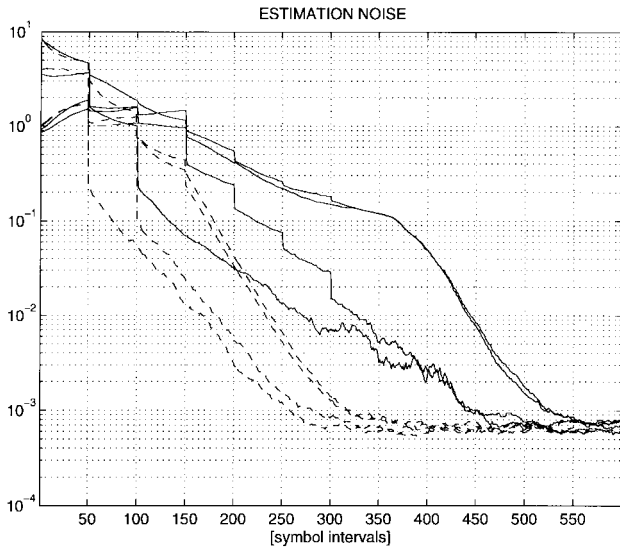


Fig. 6. Learning curves. MABW (solid) and MAV (dash-dot). Users' amplitudes: 3, 2, 1, 1. SNR = 12 dB.

curves) with the MABW algorithm is about twice that of the MAV algorithm (first 300 symbols versus 150) and that the magnitude of the jumps is smaller. Since these results are averaged over 50 runs, this reveals higher spread in the time instants where coherence is detected. To summarize, we observe that the number of parameters to be estimated is higher and that they must be extracted harmoniously; this is the reason for a lower convergence rate in the MABW case.

Second, it can be observed that despite the joint detection strategy, users are detected sequentially according to their power level (see Figs. 5 and 6). Initially, the strongest user(s) are detected by considering the rest of users as part of the noise. Later, when all users have been detected (symbol 300; see Fig. 6), the estimates of the signatures continue to improve until the floor level is reached (around symbol 550).

As shown in Fig. 6, both algorithms exhibit similar levels of estimation noise in the final steady state. In fact, when convergence has been achieved, it is absolutely equivalent to having a single *averaged* CR estimate with no contribution from states different from the most likely (MABW) or to having a very predominant state whose CR estimate will be chosen as the most likely at each instant (MAV).

The evolution of the AWGN power estimates is shown in Fig. 7. Note that both algorithms asymptotically yield the correct value for this magnitude ( $\sigma^2 = 0.063$  for a SNR equal to 12 dB).

Finally, the most important issue when working with blind algorithms is obtaining accurate enough estimates of the signatures so that, when switching to a DD algorithm, the receiver is able to finally tune such signatures or, otherwise, equalize the incoming signal [8]. In our case, this goal is successfully achieved by a set of  $M$ -tap linear equalizers. In Fig. 8, the evolution of the MSE at the output of the equalizers is plotted for all  $K$  users. In addition, such MSE's keep absolutely constant along the following 600-symbol frames, which suggests a possible earlier switch to the DD mode.

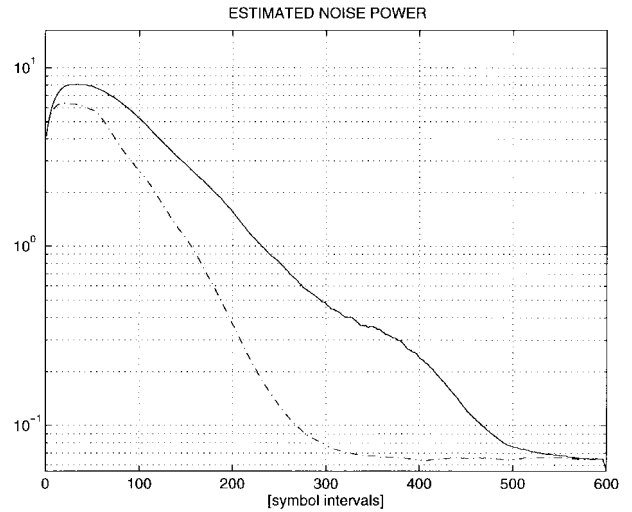


Fig. 7. Evolution of the AWGN power estimate: MABW (solid) and MAV (dash-dot). Users' amplitudes: 3, 2, 1, 1. SNR = 6 dB.

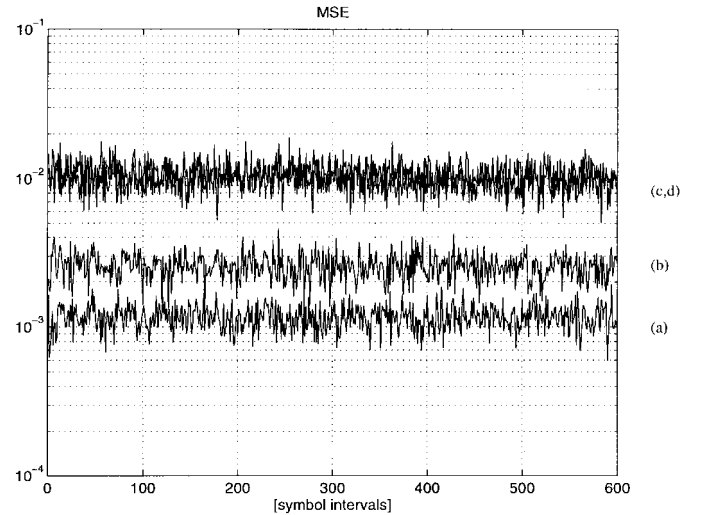


Fig. 8. Evolution of the MSE for different users observed at the output of DD equalizers. The equalizers were built from channel estimates obtained with the MAV algorithm. Users' amplitudes: (a) 3. (b) 2. (c) 1. (d) 1. SNR = 12 dB.

### B. Resistance to Channel Impairments

In Figs. 9 and 10, we plot the response of both systems in an environment with very low SNR (6 dB). Users separation is also achieved despite low SNR at the expense of a slightly slower convergence rate for the weakest users in the MABW algorithm. Of course, the remaining estimation noise is higher now, but both algorithms yield, asymptotically, the correct estimate of the AWGN power estimate ( $\sigma^2 = 0.25$ ).

Behavior when facing a strong near-far effect is depicted in Figs. 11 and 12. Approximately 30 dB power difference is considered between users, but signals are extracted within a few hundred symbols, as in the previous case. Moreover, convergence is faster now since, as long as special measures (coherence checks) were adopted to overcome the near-far effect, the difference in the received amplitudes helps the algorithm to distinguish between signals coming from different users. Consequently, CR estimation is faster.

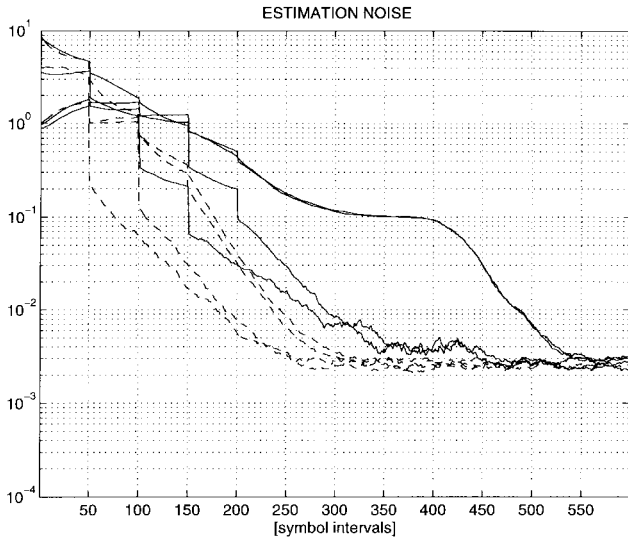


Fig. 9. Learning curves in a very low SNR environment: MABW (solid) and MAV (dash-dot). Users' amplitudes: 3, 2, 1, 1. SNR = 6 dB.

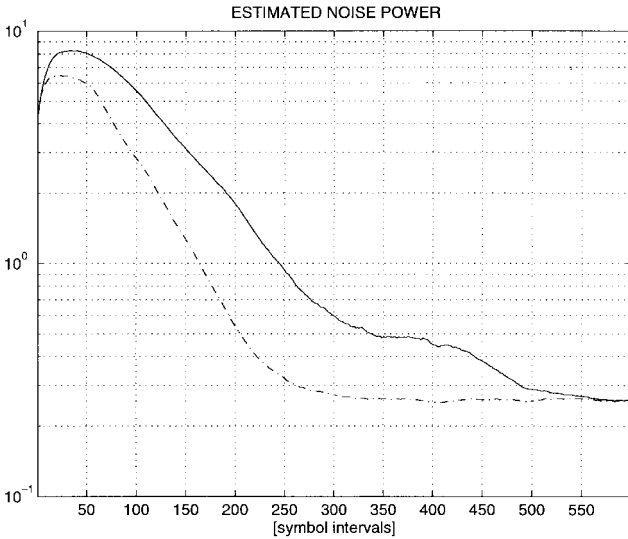


Fig. 10. Evolution of the AWGN power estimate in a very low SNR environment: MABW (solid) and MAV (dash-dot). Users' amplitudes: 3, 2, 1, 1. SNR = 6 dB.

The algorithms were also verified in the presence of large amounts of Doppler frequency shift ( $f_d T = 0.0053$ , which represents a phase shift rate of about 0.033 rads/symbol) together with severe near-far effect (20 dB) (Fig. 13). The blind DPLL is capable of tracking the evolution of a phase shift, except for an additive constant (Fig. 14), thus enabling convergence. In general, phase shift tracking is split between the blind DPLL and the CR estimate, as the latter can also track slow changes of such a parameter. In both algorithms, slower convergence can be observed due to combined effects of Doppler and near-far (Fig. 13).

To conclude this section, the algorithms are tested considering asynchronous users (Fig. 15). Delays considered (with respect to the first user) are one, two, and three chips. We do not observe significant changes in the behavior with respect to

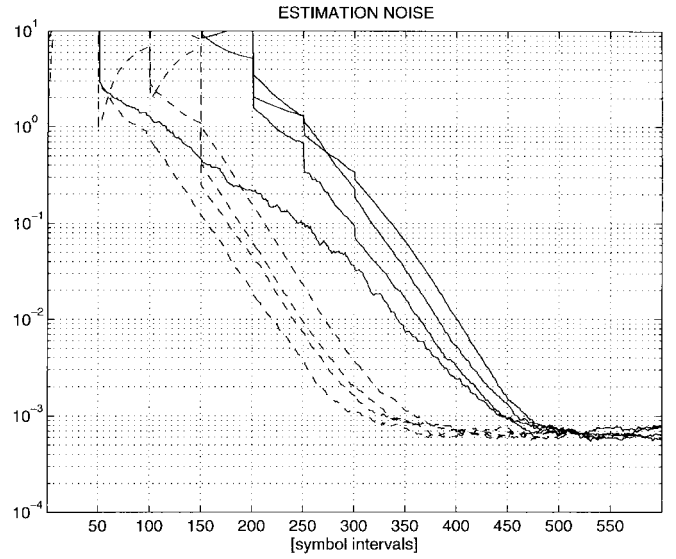


Fig. 11. Learning curves with near-far effect (30 dB). MABW (solid) and MAV (dash-dot). Users' amplitudes: 30, 7, 5, 1. SNR = 12 dB.

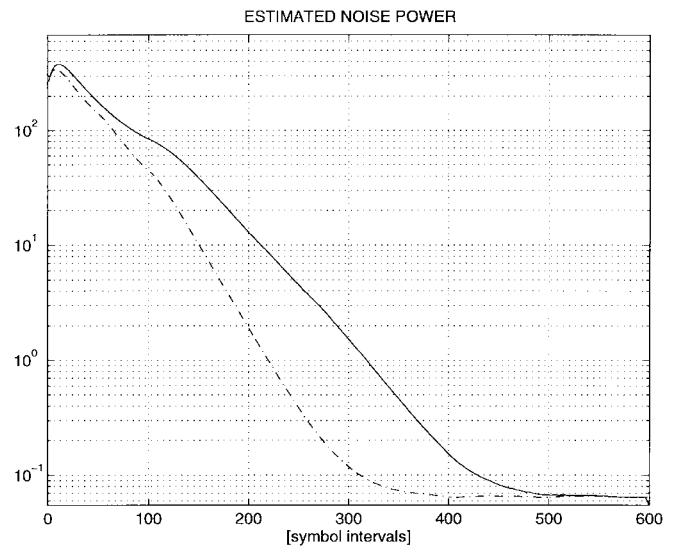


Fig. 12. Evolution of the AWGN power estimate with near-far effect (30 dB). MABW (solid) and MAV (dash-dot). Users' amplitudes: 30, 7, 5, 1. SNR = 12 dB.

the synchronous case (Fig. 6). Note, however, that the number of states is four times ( $2^2$ ) higher than before since twice the number of taps are required for the CR estimate now.

## VII. EVALUATION WITH EXPERIMENTAL DATA

Further validation of the algorithms has also been carried out by considering experimental data obtained in an underwater acoustic (UWA) environment. The UWA channel is characterized as a time-dispersive rapidly fading channel that, in addition, exhibits Doppler instabilities [31]. While vertical channels exhibit little time dispersion, horizontal channels suffer from extended multipath propagation, which usually increases with range and may cause ISI to span tens of symbol intervals. In some applications, unpredictable motion of the

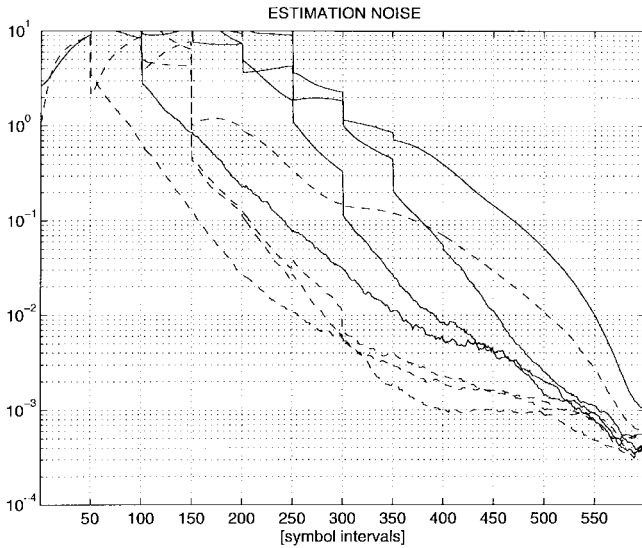


Fig. 13. Learning curves with Doppler frequency shift and near-far effect (20 dB). MABW (solid) and MAV (dash-dot). Users' amplitudes: 10, 7, 5, 1. SNR = 12 dB.

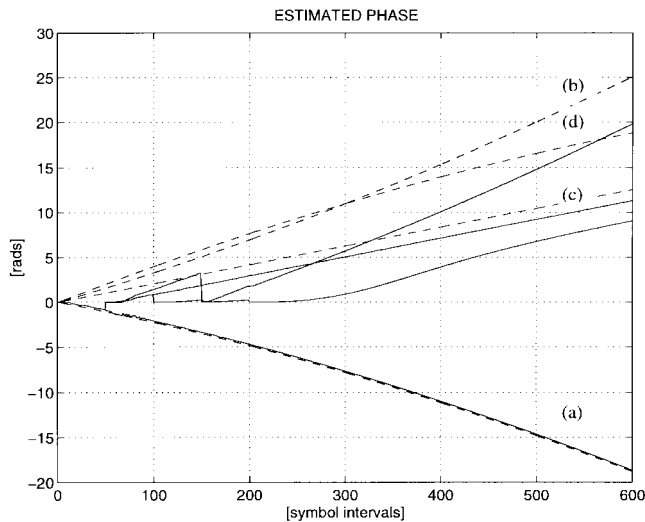


Fig. 14. Evolution of the phase shift estimates in a channel exhibiting Doppler frequency deviations and near-far effect (20 dB). MABW (solid) and true value (dash-dot). Users' amplitudes: (a) 10. (b) 7. (c) 5. (d) 1. SNR = 12 dB.

receiver and transmitter, as well as changes in the transmission medium, cause severe phase fluctuations.

Experimental data were obtained from a vertical link established between acoustic modems: The receiver was mounted on a ship, and two transmitters were deployed at a depth of 2500 m [32]. A BPSK modulation scheme was considered in which 700-symbol packets were transmitted at a data rate 1 kb/s modulating a 15-kHz carrier. Two users contributed to the CDMA signal and were assigned signatures with spreading factor 3 ( $[1, 1, 1]$  and  $[1, -1, 1]$ ) and Gaussian shaping pulses. The SNR was on the order of 20 dB, and the difference in received power between the packets was on the order of 10 dB. It is important to note that the signal was affected by severe Doppler shift deviations ( $f_d \approx 4.5$  Hz); in contrast, ISI was negligible.

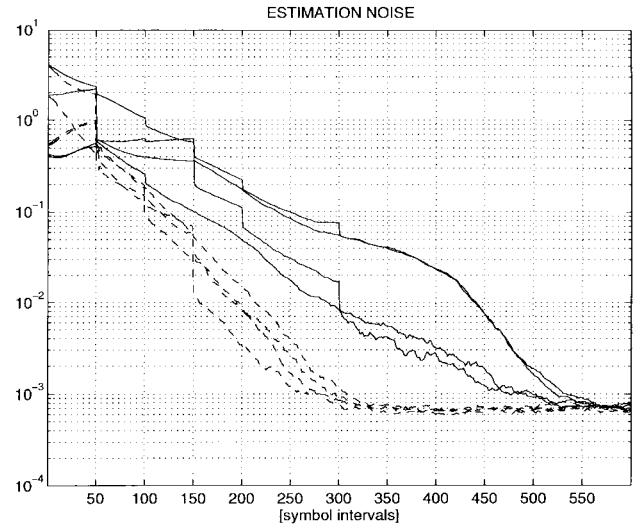


Fig. 15. Learning curves in an asynchronous case. MABW (solid) and MAV (dash-dot). Users' amplitudes: 3, 2, 1, 1. Delays with respect to the first user: 0, 1, 2, 3 chips. SNR = 12 dB.

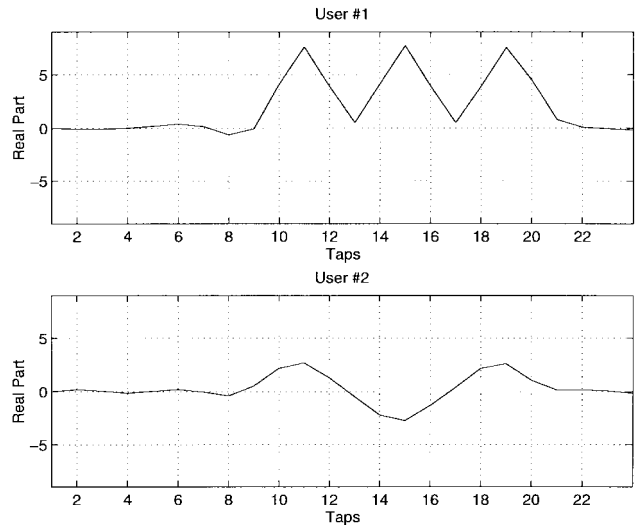


Fig. 16. MABW algorithm CR estimates for the UWA experimental-data at time instant  $n = 550$ . Users' amplitudes: 8, 3. SNR = 20 dB.

## A. Results

In order to perform channel identification, we chose the span of the estimated CR to be equal to  $L = 2$  symbols and an oversampling factor of 4 (i.e., four samples per chip). The rest of the parameters were assigned the same values as those of the preceding section. Results for the MABW algorithm are plotted in Figs. 16 and 17. First, we observe that the estimates for signatures and amplitudes corresponding to both users match the ones described above. Second, phase estimates also match, up to an additive constant, those obtained in [32], making use of training sequences (Fig. 18). Convergence is achieved within the first 150 symbols after one coherence compensation at  $n = 50$  and only 2 and 21 errors were observed, respectively, in the whole packet (700 symbols). Similar results were obtained with the MAV algorithm.

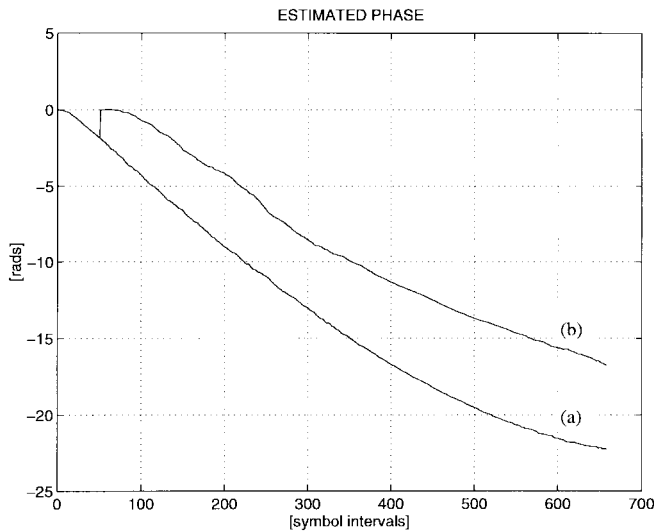


Fig. 17. Evolution of the phase shift estimated by the MABW algorithm (for both users) corresponding to the UWA experimental data. Users' amplitudes. (a) 8. (b) 3. SNR = 20 dB.

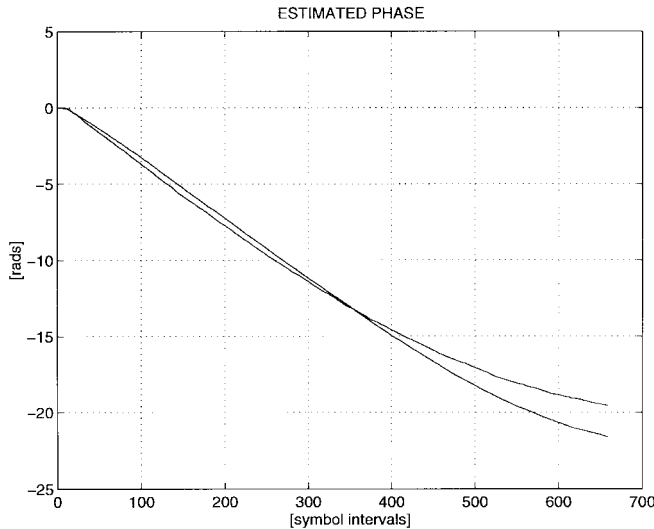


Fig. 18. Evolution of the phase shift estimated by the RLS-based nonblind algorithm proposed in [32] (for both users) corresponding to the UWA experimental data. Users' amplitudes 8 and 3. SNR = 20 dB.

### VIII. CONCLUSIONS AND FUTURE WORK

In this paper, two probabilistic algorithms for joint detection of DS-CDMA signals and channel estimation have been presented and compared. The MABW algorithm is based on the theory of the hidden Markov models, whereas the MAV is a blind version of the well-known Viterbi algorithm. Both algorithms are adaptive, and the estimate of the convolution of each user's signature with the physical channel is recursively updated using gradient schemes. Nevertheless, since the receiver operates blindly, training sequences are replaced by estimates of the transmitted data based on the received signal and the present estimate of the parameter set. The CR estimate explicitly incorporates estimates for frequency shifts obtained by a blind DPLL. The motivation for adopting such a strategy is increasing the robustness of the receiver

against large frequency deviations or phase fluctuations. Both algorithms are well suited for the startup period of a centralized multiuser receiver; after convergence, switching to any less computationally intensive DD adaptation method, such as that of MMSE, would be advisable.

As a consequence of algorithmic differences, the MABW algorithm is slightly more computationally intensive, but in contrast, memory requirements are less strict than those of MAV. Simulation study indicates that both algorithms exhibit a very similar behavior. They only differ in convergence time, which is higher in the case of MABW. In turn, this assures the convergence at least to a local maximum for this algorithm. Both algorithms are quite robust to noise, Doppler frequency shifts, asynchronism between users, and, applying the technique proposed in [20], they are near-far resistant as well.

Ongoing research mainly focuses on algorithm extension to include array observation. Computational load reduction is also being considered as well as the use of logarithmic metrics in order to avoid numerical problems associated with the conventional computation of *forward* and *backward* variables in the MABW algorithm. Further analysis is, mainly, application-specific, including synthetic signals generated by standard test channels (such as those proposed by the ETSI) and exhaustive tests with experimental data both in radio and UWA environments.

### REFERENCES

- [1] S. Verdú, "Adaptive multiuser detection," in *Proc. IEEE Intl. Symp. Spread Spectrum Techn.*, Oulu, Finland, 1994, pp. 43–50.
- [2] R. Lupas and S. Verdú, "Linear multiuser detectors for synchronous code-division multiple-access channels," *IEEE Trans. Inform. Theory*, vol. 35, pp. 123–136, Jan. 1989.
- [3] S. Verdú, "Minimum probability of error for asynchronous Gaussian multiple-access channels," *IEEE Trans. Inform. Theory*, vol. 32, no. 1, pp. 85–96, Jan. 1986.
- [4] R. S. Mowbray et al., "Increased CDMA capacity through adaptive cochannel interference regeneration and cancellation," *Proc. Inst. Elect. Eng.*, pt. I, vol. 139, pp. 515–524, 1992.
- [5] M. Honig, U. Madhow, and S. Verdú, "Blind adaptive multiuser detection," *IEEE Trans. Inform. Theory*, vol. 41, pp. 944–960, July 1995.
- [6] A. Dual-Hallen, J. Holtzman, and Z. Zvonar, "Multiuser detection for CDMA systems," *IEEE Personal Commun. Mag.*, vol. 2, pp. 46–57, Apr. 1995.
- [7] U. Madhow and M. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA," *IEEE Trans. Commun.*, vol. 42, pp. 3178–3188, Dec. 1994.
- [8] M. L. Honig, "Rapid detection and suppression of multi-user interferences in DS-CDMA," in *Proc. IEEE ICASSP*, 1995, pp. 1057–1060.
- [9] J. G. Proakis and C. L. Nikias, "Blind equalization," in *Proc. SPIE Adaptive Signal Process.*, 1991, vol. 1565, pp. 76–87.
- [10] J. Shynk, R. P. Gooch, G. Krishnamurthy, and C. K. Chan, "A comparative performance study of several blind equalization algorithm," in *Proc. SPIE Adaptive Signal Process.*, 1991, vol. 1565, pp. 102–117.
- [11] S. Verdú, B. D. O. Anderson, and R. A. Kennedy, "Blind equalization without gain identification," *IEEE Trans. Inform. Theory*, vol. 39, pp. 292–297, Jan. 1993.
- [12] J. M. Mendel, "Tutorial on higher-order statistics (spectra) in signal processing and system theory: Theoretical results and some applications," *Proc. IEEE*, vol. 79, pp. 277–305, Mar. 1991.
- [13] L. Tong, G. Xu, and T. Kailath, "Blind identification and equalization based on second order statistics," *IEEE Trans. Inform. Theory*, vol. 40, pp. 340–349, Mar. 1994.
- [14] G. B. Giannakis, "A linear cyclic correlation approach for blind identification of FIR channels," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Oct. 1994, pp. 420–424.
- [15] E. Moulines, P. Duhamel, J. Cardoso, and S. Mayrargue, "Subspace methods for the blind identification of multichannel FIR filters," *IEEE Trans. Signal Processing*, vol. 43, pp. 516–526, Feb. 1995.

- [16] M. Ghosh and C. L. Weber, "Maximum likelihood blind equalization," in *Proc. SPIE Adaptive Signal Process.*, 1991, vol. 1565, pp. 188–195.
- [17] M. Erkkut and J. G. Proakis, "Joint data detection and channel estimation for rapidly fading channels," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, 1992, pp. 910–914.
- [18] J. A. R. Fonollosa and J. Vidal, "Application of hidden Markov models to blind channel characterization and data detection," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 1994, pp. 185–188.
- [19] J. R. Fonollosa, J. A. R. Fonollosa, Z. Zvonar, and J. Vidal, "Blind multiuser identification and detection in CDMA systems," in *Proc. IEEE ICASSP*, May 1995, pp. 1876–1879.
- [20] J. R. Fonollosa, Z. Zvonar, J. A. R. Fonollosa, and J. Català, "Blind multiuser detection with array observations," to appear in *Wireless Personal Communications—Special Issue on Interference in Mobile Wireless Systems*, Issue 6, pp. 1–21, Jan. 1998.
- [21] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," *Ann. Math. Stat.*, vol. 41, no. 1, pp. 164–171, 1970.
- [22] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257–286, Feb. 1989.
- [23] M. Stojanovic, J. Catipovic, and J. G. Proakis, "Adaptive multichannel combining and equalization for underwater acoustic communications," *J. Acoust. Soc. Amer.*, vol. 94, no. 3, pp. 1621–1631, Sept. 1993.
- [24] C. Antón-Haro, J. A. R. Fonollosa, and J. R. Fonollosa, "On the inclusion of channel's time dependence in a hidden Markov model for blind channel estimation," in *Proc. IEEE Statist. Signal Array Process. Workshop*, Corfú, Greece, June 1996, pp. 164–167.
- [25] M. Seshadri, "Joint data and channel estimation using fast blind trellis search techniques," *IEEE Trans. Commun.*, vol. 42, pp. 1000–1011, Feb./Mar./Apr. 1994.
- [26] W. C. Lindsey and C. M. Chie, "A survey of digital phase-locked loops," *IEEE Trans. Commun.*, vol. COMM-69, pp. 410–431, 1981.
- [27] M. Stojanovic, J. Catipovic, and J. G. Proakis, "Phase-coherent digital communications for underwater acoustic channels," *IEEE J. Oceanic Eng.*, vol. OE-19, pp. 100–111, Jan. 1994.
- [28] F. R. Magee Jr. and J. G. Proakis, "Adaptive maximum-likelihood sequence estimation for digital signaling in the presence of intersymbol interference," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 120–124, Jan. 1973.
- [29] J. G. Proakis, *Digital Communications*, 3rd ed. New York: McGraw-Hill, 1995, pp. 849–862.
- [30] K. Dogançay and V. Krishnamurthy, "Quick aggregation of Markov chain functionals via stochastic complementation," in *Proc. IEEE ICASSP*, Munich, Germany, Apr. 1997, vol. I, pp. 63–67.
- [31] J. Catipovic, "Performance limitations in underwater acoustic telemetry," *IEEE J. Oceanic Eng.*, vol. 15, pp. 205–216, July 1990.
- [32] Z. Zvonar, D. Brady, and J. Catipovic, "An adaptive linear multiuser receiver for deep water acoustic local area networks," in *Proc. IEEE ICASSP*, Apr. 1994, vol. II, pp. 389–392.



**Carles Antón-Haro** (S'93) was born in Barcelona, Spain, in 1969. He received the M.S. degree in telecommunication engineering from the Universitat Politècnica de Catalunya (UPC), Barcelona, in 1994.

Until January 1995, he worked as a Teaching Assistant in the Department of Computer Architecture, Polytechnic School of Mataró, UPC. In 1995, he joined the Department of Signal Theory and Communications, UPC, where he is now a Research Assistant. Since 1995, he has been a recipient of a research scholarship granted by the Comissionat per

a les Universitats i Recerca (Generalitat de Catalunya) in the pursuit of the Ph.D. degree. His research interest areas include statistical signal analysis, array processing in wireless and mobile communications, channel modeling, and speech coding. Currently, he is involved in the ACTS' project technologies in smart antennas for universal advanced mobile infrastructure—Part II (TSUNAMI-II).



**José A. R. Fonollosa** (M'86) was born in Barcelona, Spain, on April 19, 1963. He received the M.S. and Ph.D. degrees in telecommunication engineering from the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, in 1986 and 1989, respectively.

Since 1986, he has been with the Department of Signal Theory and Communications, UPC, where he is now an Associate Professor. From August 1991 to July 1992, he was visiting the Signal and Image Processing Institute, University of Southern California, Los Angeles. His research interest areas include statistical signal analysis and its applications in communication systems and speech processing algorithms.

Dr. Fonollosa received the 1992 Marconi Youngest Scientist Award. He is a member of the European Speech Communication Association and the Acoustical Society of America. He served as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING June 1992 to June 1995.



**Zoran Zvonar** was born in April 1962, in Belgrade, Yugoslavia. He received the Dipl. Ing. degree in 1986 and the M.S. degree in 1989, both from the Department of Electrical Engineering, University of Belgrade, and the Ph.D. degree in electrical engineering from Northeastern University, Boston, MA, in 1993.

From 1986 to 1989, he was with the Department of Electrical Engineering, University of Belgrade, where he conducted research in the area of telecommunications. From 1993 to 1994, he was a postdoctoral investigator at the Woods Hole Oceanographic Institution, Woods Hole, MA, where he worked on multiple-access communications for underwater acoustic local area networks. Since 1994, he has been working on the design of wireless communications systems, with emphasis on the integrated solutions for various digital standards, including GSM and DECT. He is also Adjunct Professor at the Department of Electrical and Computer Engineering, Northeastern University. His research interests include theory and implementation issues of wireless communications systems, multiuser detection, and estimation and spread spectrum communications.



**Javier R. Fonollosa** (S'90–M'92–SM'98) was born in Barcelona, Spain, on February 20, 1966. He received the telecommunication engineering degree with highest honors from the Universitat Politècnica de Catalunya (UPC), Barcelona, in 1988 and the Ph.D. degree in electrical and computer engineering from Northeastern University, Boston, MA, in 1992.

In 1989, he held a Visiting Appointment at the European Space Agency Center, Amsterdam, The Netherlands. From the Fall of 1989 until 1992, he was Research Assistant at Northeastern University, and the University of Southern California, Los Angeles, where, on completion of the Ph.D. degree, he became a Research Associate. In 1993, he joined the Department of Signal Theory and Communications, UPC, where he became Associate Professor in November 1996. His research interests include statistical signal processing with emphasis on array signal processing and multiuser communications.

Dr. Fonollosa is the author of eight journal and 30 conference papers in the areas of signal processing and communications. Since 1994, he has participated in the European Commission funded projects TSUNAMI, TSUNAMI-II, and SUNBEAM, which include the deployment of adaptive antennas in second- and third-generation cellular mobile communication systems. In June 1995, he was co-chairman and organizer of the IEEE Signal Processing/ATHOS Workshop on Higher-Order Statistics, Girona, Spain.